

REMARKS

Responsive to the telephone call with the Examiner of May 23, 2003, please kindly enter the
5 following Amendment After Final Office Action mailed on April 1, 2003.

Although the Examiner declined, on May 23, 2003, a request for an Interview due to his heavy
workload, Applicant respectfully requests a reconsideration of the decision made in the Final
Office Action and an Interview with the Examiner to be held after July 6, 2003. Final Office
10 Action is premature because there is obvious misunderstanding of the present invention and prior
art and the Applicant would like to have an opportunity to come to a meeting of the minds and
place the application in condition for allowance, if a new amendment is requested by the
Examiner.

15 1. Reconsideration, withdrawal of the final rejection and continued examination is respectfully
requested because the Applicant opines that all claims are in good condition for allowance. The
final rejection of all claims was premature because new grounds for rejection which were not
necessitated by amendment of claims, which were only made in order to define the invention
more clearly.

20 2. The Examiner stated that Applicant's comments from Amendment page 9, lines 25-27 and
page 10, lines 7-8 are not recited in the rejected claims, although he interpreted them in light of
the specification.

25 Applicant respectfully disagrees with this opinion and reiterates that the quoted language exists
in the claims, as previously amended, in preamble and step (b). It is obvious from the claims, as
filed, that they are directed to "pre-processing" of an existing access code. Step (b) is hereby
further amended to place quote from the preamble inside step (b).

3. The specification was objected to as failing to provide proper antecedent basis for the claimed subject matter. It was stated that the limitation "to provide a direct call mechanism", added in the claims 1, 10 and 19, was not specified in the specification.

5 Applicant respectfully points to thirteen locations in the Specification where the direct call is described and named as a "call": page 3, lines 3, 6, 9 and 11; page 7, lines 7, 13, 16 and 21; page 8, lines 4, 5, 7, 8 and 31. Term "direct call" appears in the Title of the invention and in the Abstract. Because a call is used to prevent a need for a lookup, it is a "direct" call, obtained via pointer.

10

4. Examiner stated that the recitation in claims 1, 10 and 19 has not been given patentable weight because it occurs in the preamble ("a direct call mechanism"), merely recites the purpose of a process or the intended use of a structure and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone.

15

Applicant respectfully points out that not only the preamble but also the element (b) in claims 1, 10 and 19, as previously amended, states "to provide a direct call mechanism". Moreover, a software "call" is neither a "purpose of a process" nor "intended use of a structure". This term is only added for clarification because it was clear from the First Office Action that claims and specification were not understood.

20

Applicant appreciates Examiner's comment about preamble with gratitude and further amends step (b) of the independent claims by adding some language from the preamble for further clarification of independent claims.

25

5. Claims 1-22 are pending in this application. All independent claims 1, 10 and 19 have been amended in order to further clarify the present invention. Claim 1 has been amended to state:

1. A method for **pre-processing an access plan** generated for a query in a relational database management system to include a direct call mechanism replacing a lookup function of a run-time interpreter, said access plan including a plurality of operation codes, each of said operation codes being associated with one or more executable functions for performing the query, said method comprising the steps of:

(a) determining from the access plan an executable function associated with a first operation code; and

(b) augmenting said first operation code in the access plan with a pointer to said executable function to provide a direct call mechanism replacing a lookup function of a run-time interpreter.

As can be see from the bolded terms, claimed method is for **pre-processing** of an existing access plan which has op. codes and is to be interpreted by a Software Interpreter (not a Compiler). Step (a) determines for an op. code associated executable function which should be interpreted if there is no present invention. Step (b) augments the op. code, inside the existing access plan, and disallows interpretation of the executable function by the Interpreter because it replaces the op. code with a pointer to a function, located outside the access plan, which implements the operation indicated by the op. code or an intermediate function, thus removing the interpretive step. This means that, when the access plan is being executed at **run-time**, instead of interpreting the op. code, the pointer is used to **call** the replacement code, thus a term "direct call", which is described in much more detail in the Specification.

Thus, the claims are directed to a run-time improvement of execution of an existing and previously optimized access plan whose steps have already been determined and are not changed by the present invention but its low level operations are being replaced with other low level operations to increase the run-time speed.

However, Boykin reference is from a different field and performs a different function. It is directed to a method used to **generate a new**, optimized access plan. A **Compiler**, at compilation time, replaces rules for enforcing constraints inside the access plan with the actual code that enforces the constraints, so that all the code is located **inside** the access plan. Present invention
5 does not change the access plan steps but uses a pointer to an external code. It is true that Boykin reference also uses pointers. A pointer is an often-used software tool which points to a code or data, depending on its function. In Boykin reference a pointer points to the code added **inside** the access plan. Pointers in the present invention point to a function located **outside** the access plan.

10 Rejections of claims 1-22 under 35 USC Sec. 102 and 103 are now moot because the Applicant amended the claims in order to define the invention more particularly. All independent claims 1, 10 and 19 are now specifically directed to show an improvement of a standard database management system which includes a direct call mechanism replacing the lookup function of a run-time interpreter and a method for pre-processing an access plan to provide a direct call
15 mechanism in such a system. They recite novel structure and thus distinguish over the cited prior art, under 35 U.S.C. 102(e) and 103.

7. Claims 1-3, 6-7, 10-12, 15-16 and 19 stand rejected under 35 U.S.C. 102(e) as being
20 anticipated by US Patent No. 5, 386,557 issued to Boykin et al. (herein Boykin).

The Examiner stated that, with respect to claim 1, Boykin discloses (a) determining from the access plan an executable function associated with a first operation code (access plan contains low-level interpreted code which has been compiled from database access language such as SQL and the thread, item 50, containing code for performing the function of the SQL
25 statement (see fig. 1, and fig. 2, col. 3, lines 60-67 and col. 4, lines 20-25; also see col. 5, lines 35-49); and (b) augmenting said first operation code in the access plan with a pointer to said executable function (item 54 IUDO containing pointers to any data values being necessary to perform the execution of the operations: col. 4, lines 44-65 and col. 6, lines 36-42).

The Examiner stated that, with respect to claim 2, Boykin discloses the remaining operation codes in the access plan (col. 5, lines 35-62).

5 The Examiner stated that, with respect to claim 3, Boykin discloses augmenting said first operation code in the access plan with a pointer to an intermediate function, said intermediate function including a data structure for storing a pointer to said executable function (col. 4, lines 20-32 and lines 44-65).

10 The Examiner stated that, with respect to claim 6, Boykin discloses assessing the executable function associated with the first operation code and if applicable, replacing the call to the executable function with a call to a second executable function (col. 9, lines 2-7).

The Examiner stated that, with respect to claim 7, Boykin discloses intermediate function which includes processing operations for the first operation code or the executable function associated with the first operation code (col. 4, lines 44-65).

15 The Examiner stated that claim 10 is essentially the same as claim 1 except that it is directed to a computer program product rather than a method (see fig. 1, and fig. 2, col. 3, lines 60-67 and col. 4, lines 20-25; also see col. 5, lines 35-49; and item 54 IUDO containing pointers to any data values being necessary to perform the execution of the operations: col. 4, lines 44-65 and col. 6, lines 36-42), and is rejected for the same reason as applied to the claim 1 hereinabove.

20 The Examiner stated that claim 11 is essentially the same as claim 2 except that it is directed to a computer program product rather than a method (col. 5, lines 35-62), and is rejected for the same reason as applied to the claim 2 hereinabove.

25 The Examiner stated that claim 12 is essentially the same as claim 3 except that it is directed to a computer program product rather than a method (col. 4, lines 20-32), and is rejected for the same reason as applied to the claim 3 hereinabove.

The Examiner stated that claim 15 is essentially the same as claim 6 except that it is directed to a computer program product rather than a method (col. 9, lines 2-7), and is rejected for the same reason as applied to the claim 6 hereinabove.

5 The Examiner stated that claim 16 is essentially the same as claim 7 except that it is directed to a computer program product rather than a method (col. 4, lines 44-65), and is rejected for the same reason as applied to the claim 7 hereinabove.

The Examiner stated that claim 19 is essentially the same as claim 1 except that it is directed to a system rather than a method (see fig. 1, and fig. 2, col. 3, lines 60-67 and col. 4, lines 20-25; also see col. 5, lines 35-49; and item 54 IUDO containing pointers to any data values
10 being necessary to perform the execution of the operations: col. 4, lines 44-65 and col. 6, lines 36-42), and is rejected for the same reason as applied to the claim 1 hereinabove.

The claims of the present invention recite novel structure and thus distinguish over the cited prior art, under 35 U.S.C. 102(e). They specifically describe an improved database management
15 system and method capable of providing a direct call mechanism, thus replacing the existing lookup function of a run-time interpreter, as described in Specification on p. 2, li. 16-31; p. 3, li. 2-10, p. 6, li. 14-30, pages 7-9, to provide faster access which is cost-effective. These features are recited in claims 1-22 and throughout the Abstract and Specification, including Summary, and are not shown in the Boykin reference which does not show all the features of the present
20 invention, operating in the same way and for the same purpose.

Reference Boykin teaches a system for enforcing referential constraints when changes are made to a database, where the RDBMS does not interpret database access commands at run time (col 1, li. 19-23). Moreover, according to col. 1, li. 55-68, the code in Boykin reference is inserted
25 directly into the access plan at compilation time. Therefore, Boykin reference does not have the steps (a) - (b) of the independent claims of the present invention. It is from a completely different field, has different components, works in a different mode of use and produces different results. Boykin reference does not show, teach or suggest an improvement of a standard database management system and a method for replacing a lookup function of a run-time interpreter with

a direct call mechanism that integrates easily into an existing RDBM system and is cost-effective, which was a long-felt need in this field.

5 The Examiner stated that, as per claim 1-3, 6-7, 10-12, 15-16 and 19, the Boykin reference in cols. 3-6 and 9 shows augmenting of an access plan to IUDO pointers. However, these columns and col 10 li. 9-17 in Boykin describe a feature wherein the IUDOs for inserts, deletes and updates are made at the time the access plan is compiled and the appropriate subthreads are inserted by the compiler, as shown in col. 5, li. 5-12 and 35-56. Such complexity is not needed in the present invention and this technique cannot be applied to it. Moreover, these lines teach away
10 from the present invention which does not compile the code but pre-processes the access plan to provide a direct call at run-time, the feature that does not exist in Boykin reference.

Therefore, all independent claims 1, 10 and 19 and all claims dependent upon these in the present invention recite novel structure and thus distinguish over the cited prior art, Boykin, and are not
15 anticipated by it under 35 U.S.C. 102(e).

8. Claims 4-5, 8-9, 13-14, 17-18 and 20-22 stand rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 5,386,557 issued to Boykin et al. (herein Boykin) in view of US Patent No. 5,822,750 issued to Jou et al. (herein Jou).

The Examiner stated that, with respect to claims 4-5, Boykin discloses the method as
20 discussed in claim 1. It is noted with appreciation that the Examiner held that Boykin does not explicitly indicate, "data structure includes means for storing information and data structure providing means for storing information" but that Jou discloses data structure for storing information as claimed (col. 10, lines 10-67, col. 11, lines 1-14 and also see col. 3, lines 20-36) and that, therefore, it would have been obvious to a person of ordinary skill in the art at the time
25 the invention was made to combine the teachings of Boykin with the teachings of Jou so as to obtain a method for pre-processing access plan from a query, and that this combination would provide a method enabling to optimize derived table evaluation and to eliminate the table access operations in evaluation of a query increases efficiency (Jou - col. 8, lines 60-65) in the relational database management environment.

The Examiner stated that, with respect to claims 8-9, Boykin discloses the method as discussed in claim 1. It is noted with appreciation that the Examiner held that Boykin does not explicitly indicate, "gathering statistics on the use of the executable function; and a pause for receiving user input before or after the call to the executable function" and that Jou discloses gathering statistics data (col. 11, lines 38-51; also col. 12, lines 32-67 and col. 13, lines 1-15) and receiving user input as claimed (col. 12, lines 18-45) and that, therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Boykin with the teachings of Jou so as to obtain a method for pre-processing access plan from a query and that this combination would provide a method enabling to optimize derived table evaluation and to eliminate the table access operations in evaluation of a query increases efficiency (Jou - col. 8, lines 60-65) in the relational database management environment.

The Examiner stated that claims 13-14 are essentially the same as claims 4-5 except that it is directed to a computer program product rather than a method (col. 10, lines 10-67, col. 11, lines 1-14 and also see col. 3, lines 20-36), and are rejected for the same reason as applied to the claims 4-5 hereinabove.

The Examiner stated that claims 17-19 are essentially the same as claims 8-9 except that it is directed to a computer program product rather than a method (col. 11, lines 38-51; also col. 12, lines 32-67 and col. 13, lines 1-15; and col. 12, lines 18-45), and are rejected for the same reason as applied to the claims 8-9 hereinabove.

The Examiner stated that, with respect to claims 20-22, Boykin discloses the method as discussed in claim 19 and that it discloses adding another pointer (see fig 2, col. 4, lines 20-34 and 6, 36-42). It is noted with appreciation that the Examiner held that Boykin does not explicitly indicate, "replacing said operation codes; storing information and data structure" and that Jou discloses replacing operation as claimed col. 14, lines 26-45); data structure for storing information as claimed (col. 10, lines 10-67, col. 11, lines 1-14 and also see col. 3, lines 20-36)

and that, therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Boykin with the teachings of Jou so as to obtain a method for pre-processing access plan from a query and that this combination would provide a method enabling to optimize derived table evaluation and to eliminate the table access operations in evaluation of a query increases efficiency (Jou - col. 8, lines 60-65) in the relational database management environment.

9. All claims in the application have joint inventors.

10. Claims 4-5, 8-9, 13-14, 17-18 and 20-22 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Boykin in view of Jou. However, since it has been shown above that all independent claims 1, 10 and 19 and all claims dependent upon these in the present invention recite novel structure and thus distinguish over the cited prior art, Boykin, and are not anticipated by it under 35 U.S.C. 102(e), the combination of Boykin and Jou does not make the present invention obvious and unpatentable under Section 103 for the same reasons stated above.

Jou reference is directed to optimization of correlated queries and has nothing in common with the present invention, according to the Examiner, except the data structure for storing information and gathering statistic data. Thus, neither Jou nor Boykin teach improving a RDBMS by replacing a lookup function of a run-time interpreter with a direct call mechanism.

Therefore, the present invention is created to solve problems from a different art field than Boykin and Jou. Moreover, Boykin and Jou references do not perform any elements of the independent claims 1, 10, and 19, and their dependent claims. Therefore, the combination of Boykin and Jou fails to teach all elements of claims 1, 10 and 19 and their dependent claims.

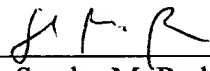
Moreover, Boykin and Jou satisfy a different need from a different area and do not teach optimization during run time. Therefore, Boykin and Jou references cannot be used to invalidate independent claims 1, 10, and 19 and their dependent claims.

Therefore, all submitted claims are allowable over the cited reference and their reconsideration is respectfully requested. None of the cited references discloses the subject matter and features of claims 1-22 of the present invention and, even if they did show some individual features, they would not be able to meet the claims of the present invention which provide new and unexpected results over these references and are thus not anticipated under Section 102 and unobvious and patentable under Section 103.

In view of the above, it is submitted that this application is now in good order for allowance, which applicant respectfully solicits. Should matters remain which the Examiner believes could be resolved in a telephone interview, the Examiner is kindly requested to telephone the applicant's undersigned attorney. No additional fee is required in connection with this communication since the Amendment is mailed within three months from the Office Action and the number of claims is not extending the original number of claims. However, any underpayment is authorized to be charged to Deposit Account Number **09-0460** in the name of IBM Corporation.

Respectfully submitted,

Date: June 4, 2003



Sandra M. Parker
Reg. No. 36,233

LAW OFFICE OF SANDRA M. PARKER
329 La Jolla Avenue
Long Beach, CA 90803
Phone: (562) 597-7504
FAX: (562) 597-1841.